# Automating RF Circuit Synthesis

Anirudh B S (EE21B019) Sankaran Aniruddhan

***Abstract*—With the advent of Machine Learning algorithms in almost every aspect of life, it has become all the more important to make efficient use of automation in electronics used in everyday life. The use of AI and ML in the VLSI Industry has been on the rise in the recent past. In this paper, we will be automating the design of the Noise Cancelling LNA and the Resistive Feedback LNA subject to a given objective function. We will be employing Gradient Descent in achieving the same.**

***Index Terms*—Noise Cancelling LNA, Resistive Feedback LNA, CAD, Gradient Descent, optimization**

## I. Introduction

Analog circuits play an important role in processing real world signals and interfacing them with digital signal processing units. The design of analog circuits is challenging as the parameters of the analog circuits are interrelated and difficult to tune to achieve the desired performance. Optimizing one parameter can result in the degradation of another. Designers are always trying to achieve an optimum trade-off between the performance metrics. Analog circuits have non-linearities associated with their components. Hence, the final optimum design of a circuit takes a large number of iterations for the fine-tuning of the circuit parameters. [2], [4].

Low noise amplifier (LNA) is a first component of any receiver front end circuitry, on which receiver performance depends. It is the first amplifying block of front end. Since the signals received from antenna is very weak $1\mu$V. LNA is first gain stage, which boosts the weak input signal from antenna. Since the demand of wireless communication is at high pace, the optimum design of its component becomes major part. Along with amplification it has a significant impact on the noise performance of receiver circuitry. As the result we get a significant signal without noise dominating it. [5].

In recent years, a lot of effort is made to obtain optimally designed LNA in terms of Noise Figure as lowest possible and linearity to be maximum possible. The scaling of channel length in VLSI process of MOS has reduced to nanometers region, result raising the transit frequency gigahertz. It has also helped in reducing power consumption (milli watts). The challenges are there and imply motivation in exploration of RF Architectures.

With the advent of Machine Learning algorithms, it has become easier for an RF engineer to exploit the already available optimization techniques to minimize power dissipation while meeting the desired specifications. In this paper, we shall be employing gradient descent in finding the minimum power dissipated in a Resistive Feeback LNA [3] and Noise Cancelling LNA topologies [1] while meeting the desired specifications.

## II. Optimization Algorithm

The Gradient Descent Algorithm has been used in order to converge to the local minima of the objective function. The objective function is a linear combination of several different objective functions specific to gain, $IIP_3$, NF, $S_{11}$ and $I_{dd}$ defined in the following manner

Gain Loss = ReLU(Gain specification - Simulation Gain)

$IIP_3$ Loss = ReLU($IIP_3$ specification - Simulation $IIP_3$)

NF Loss = ReLU(Simulation NF - NF specification)

$S_{11}$ Loss = ReLU(Simulation $S_{11}$ - $S_{11}$ specification)

$I_{dd}$ Loss = ReLU($I_{dd}$ specification - Simulation $I_{dd}$)

where the ReLU activation is defined as

$$ReLU(x) = max\{x, 0\} \tag{1}$$

The notation of defining Gain Loss as Gain specification - Simulation Gain is obvious from the reason that the loss contributed by that term should go to zero if the simulation gain exceeds the gain specification. Similarly, if the simulation $S_{11}$ is lower than the $S_{11}$ specification, the loss contributed by that term also needs to go to zero. Thus, the above convention has been used. Unless otherwise mentioned elsewhere, $I_{dd}$ specification has been by definition assumed to be 0 since we would ideally want power dissipated to be zero. The total loss is defined as the following

Loss = $\alpha_1$ Gain Loss + $\alpha_2$ $IIP_3$ Loss + $\alpha_3$ NF Loss + $\alpha_4$ $S_{11}$ Loss + $\alpha_5$ $I_{dd}$ Loss

This algorithm works for any generic topology, only the gradients need to be modified depending on the topology being considered. The mismatch component has been added to the code dealing with the sensitivity and the PVT check.

# III. Noise Cancelling LNA

## A. Schematic

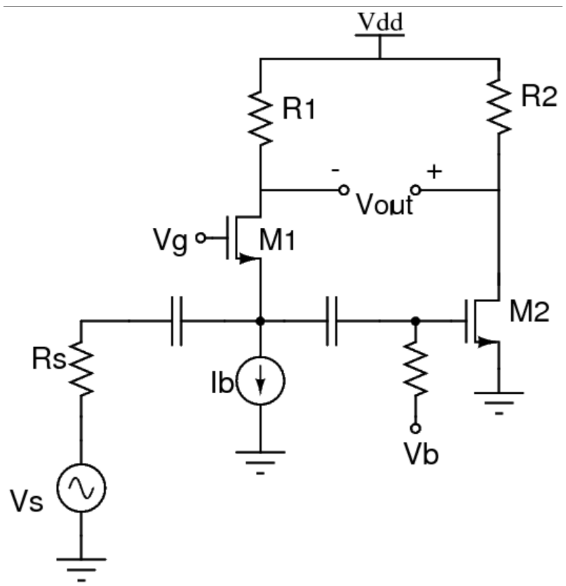


Fig. 1. Noise Cancelling LNA Schematic

The above schematic (Fig 1) was been built on Cadence Virtuoso and the DC Operating Point was set such that the input impedance looking in from the Input Port (PORT0) is 50Ω. The TSMC28 nm PDK file was used with the predefined sizes. To change the width, the multiplier was modified accordingly by the code. The operating frequency is 1 GHz. The biasing mode used is Negative Feedback biasing due to its robustness with respect to parameter variations. The Common Gate Stage has been biased using a current source while the Common Source Stage has been biased using a current mirror. The DC Operating Point was set such that the $g_m$ of the common gate stage was close to 20 mS (21.1 mS) such that the input impedance specification is met. The results have been summarized in Table I. The designer needs to ensure that the noise cancellation condition is met at the time of the schematic design.

## B. Small Signal Analysis

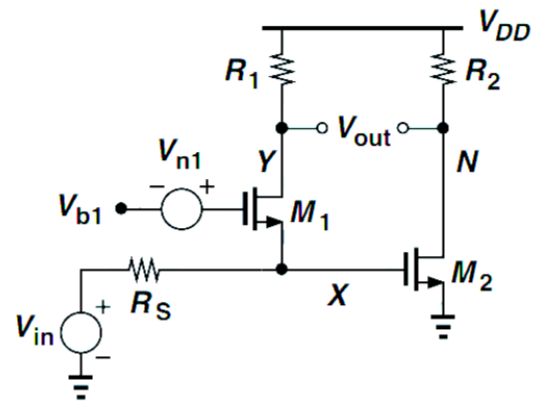


Fig. 2. Small Signal Model of the Noise Cancelling LNA
Ref. RF Microelectronics, Razavi

### 1) $S_{11}$ Expression:

$$Z_{in} = \frac{1}{g_{m1}} = R_s = 50\Omega \tag{2}$$

$$S_{11} = \frac{Z_{in} - R_s}{Z_{in} + R_s} \tag{3}$$

*2) Condition for Noise Cancellation:* The condition for noise cancellation is given by equation (4)

$$g_{m1} R_1 = g_{m2} R_2 \tag{4}$$

*3) Gain Expression:* The expression for the small signal differential output to single ended input gain is given by

$$\frac{V_{out}}{V_{in}} = g_{m1} R_1 = g_{m2} R_2 \tag{5}$$

*4) Expression for Noise Figure:* The Noise Figure is given by the following expression

$$NF = 1 + \frac{R_s}{R_1} + \gamma \frac{R_2}{R_1} + \frac{R_s R_2}{R_1^2} \tag{6}$$

where $\gamma$ is the excess noise factor which is equal to $\frac{2}{3}$ for long channel MOSFETs. The above set of equations provides a starting point for the simulation, the hand calculated values came out to be $m_s$=20, $m_g$=80, $R_2$=500 Ω, $R_1$=200 Ω, $I_{b2}$=900uA and $I_{b1}$=10uA. The results have been presented in the following subsection.

## C. Update Equations

If Gain Loss $> 0$:

$$R_1 \leftarrow R_1 + \alpha_1; R_2 \leftarrow R_2 + \alpha_1$$

If NF Loss $> 0$:

$$R_1 \leftarrow R_1 - \alpha_2; R_2 \leftarrow R_2 - \alpha_2$$

If $I_{dd}$ Loss $> 0$:

$$m_s \leftarrow m_s + \alpha_1; m_g \leftarrow m_g + \alpha_1$$

If $S_{11}$ Loss $> 0$:

$$m_g \leftarrow m_g + \alpha \frac{20}{m_g - 5}$$

If $IIP_3$ Loss $> 0$:

$$m_s \leftarrow m_s - \alpha_2; m_g \leftarrow m_g - \alpha_2$$

where $\alpha$, $\alpha_1$ and $\alpha_2$ are tunable hyperparameters called the **learning rate**(s).

### *D. Results*

The code was run for 100 iterations and the best performing netlists were handpicked. By best performing netlists, here I mean that the netlists whose loss function was less than a specified threshold. This threshold was determined by running the code several times. This is a **tunable hyperparameter** of the model used. Out of all the best performing netlists, the netlist that had the least sensitivity was handpicked and its properties have been listed in Table I.

| Requirement | Specification | Starting Point | Final Result |
|---|---|---|---|
| Gain | 15 dB | 15.82 dB | 18.85 dB |
| Noise Figure | 5 dB | 5.107 dB | 2.938 dB |
| $S_{11}$ | -10 dB | -17.5354 dB | -14.51 dB |
| $IIP_3$ | -10 dBm | -7.08 dBm | -5.22 dBm |
| Power | Minimize | 1.322 mW | 1.79 mW |
| Loss | | 2.108 | 0.00199 |

TABLE I
NOISE CANCELLING LNA - RESULTS

The temperature analysis for the best netlist is given below in Table II.

| Temperature | Loss |
|---|---|
| -40°C | 0.001938817 |
| 0°C | 0.00197025 |
| 27°C | 0.001990783 |
| 100°C | 1.3920957 |

TABLE II
TEMPERATURE VARIATIONS

The process analysis for the best netlist is given below in Table III.

| Process | Loss |
|---|---|
| TOP TT | 0.001990783 |
| TOP FF | 0.002053277 |
| TOP FS | 0.002039875 |
| TOP SF | 0.07775 |
| TOP SS | 0.139516461 |

TABLE III
PROCESS VARIATIONS

The supply voltage analysis for the best netlist is given below in Table IV

| Supply Voltage | Loss |
|---|---|
| 0.81 V | 0.00689424 |
| 0.90 V | 0.001990783 |
| 0.99 V | 0.002081644 |

TABLE IV
SUPPLY VOLTAGE VARIATIONS

The sensitivity was evaluated to be 1.392x10$^{-5}$ while the temperature sensitivity evaluated to be 2.04x10$^{-7}$. The time taken to run this code was about 35 minutes.
The parameters of the best netlist are as $m_s$=119.0, $m_g$=179.0, $R_2$=500 $\Omega$, $R_1$=200 $\Omega$, $I_{b2}$=900uA and $I_{b1}$=9.99uA
The number of netlists that passed the Gradient Descent Loss Threshold were 68, the number of netlists that passed the PVT check were 32 and the number of netlists that passed the sensitivity check were 10 over 100 iterations. The lowest loss obtained was 0.00199 and sensitivity of the same was 1.392x10$^{-5}$.

### *E. Loss Function*

As anticipated the loss function kept on decreasing as the number of iterations increased. This indicates that the algorithm used is converging to the local minimum solution as required. The loss function used in this case was the simple Rectified Linear Unit (ReLU) loss function defined in equation (1) The total loss was a sum total of the losses corresponding to the Gain, $S_{11}$, $IIP_3$, $I_{dd}$ and Noise Figure (NF) as described in Section II.

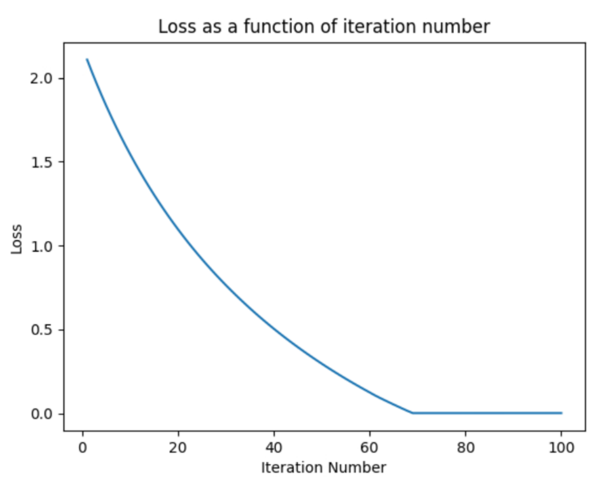


Fig. 3. Loss progression in Noise Cancelling LNA

## IV. Resistive Feedback LNA

### A. Schematic

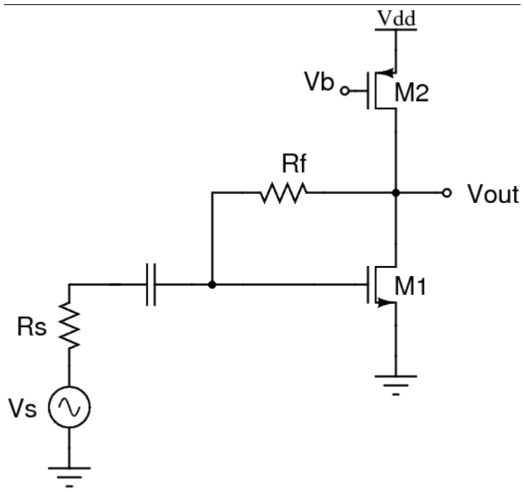


Fig. 4. Resistive Feedback LNA Schematic

The above schematic (Fig 4) was been built on Cadence Virtuoso and the DC Operating Point was set such that the input impedance looking in from the Input Port (PORT0) is 50Ω. The TSMC28 nm PDK file was used with the predefined sizes. To change the width, the multiplier was modified accordingly by the code. The operating frequency is 1.58 GHz. The biasing mode used is Negative Feedback biasing due to its robustness with respect to parameter variations. The input nMOSFET has been inductively degenerated to match the input impedance to 50Ω. The DC Operating Point was set such that the $g_m$ of this nMOS was close to 20 mS (20.8 mS) such that the input impedance specification is met. The results have been summarized in Table V.

### B. Small Signal Analysis

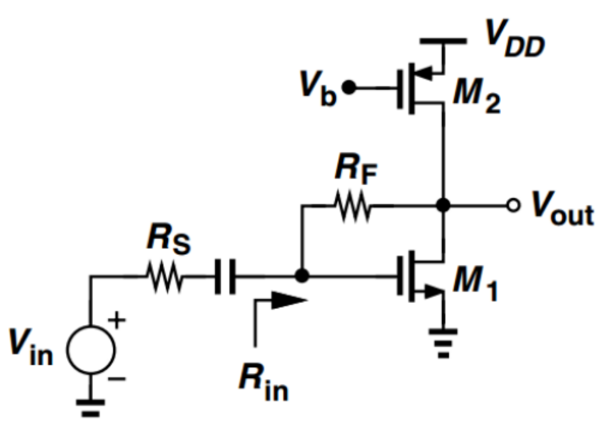


Fig. 5. Small Signal Model of the Resistive Feedback LNA
Ref. RF Microelectronics, Razavi

#### 1) $S_{11}$ Expression:

$$Z_{in} = \frac{1}{g_{m1}} = R_s = 50\Omega \tag{7}$$

$$S_{11} = \frac{Z_{in} - R_s}{Z_{in} + R_s} \tag{8}$$

*2) Gain Expression:* The expression for the small signal differential output to single ended input gain is given by

$$\frac{V_{out}}{V_{in}} = \frac{1 - g_{m1}R_f}{1 + g_{m1}R_s} \approx -\frac{R_f}{R_s} \tag{9}$$

*3) Expression for Noise Figure:* The Noise Figure is given by the following expression

$$NF \approx 1 + 4\frac{R_s}{R_f} + \gamma + \gamma g_{m2}R_s \tag{10}$$

where $\gamma$ is the excess noise factor which is equal to $\frac{2}{3}$ for long channel MOSFETs. The above set of equations provides a starting point for the simulation, the hand calculated values came out to $m_1$=20, $m_2$=80, $R_f$=300 Ω, $I_b$=10uA, $L_s$ = 1pH and $L_g$ = 100nH. The results have been presented in the following subsection.

### C. Update Equations

If Gain Loss $> 0$:

$$R_f \leftarrow R_f + \alpha_1; m_1 \leftarrow m_1 + \alpha_1$$

If NF Loss $> 0$:

$$R_f \leftarrow R_f - \alpha_2$$

If $I_{dd}$ Loss $> 0$:

$$m_1 \leftarrow m_1 + \alpha_1; m_2 \leftarrow m_2 + \alpha_1$$

If $S_{11}$ Loss $> 0$:

$$m_1 \leftarrow m_1 + \alpha\frac{20}{m_g - 5}$$

If $IIP_3$ Loss $> 0$:

$$m_s \leftarrow m_s - \alpha_2; m_g \leftarrow m_g - \alpha_2$$

where $\alpha$, $\alpha_1$ and $\alpha_2$ are tunable hyperparameters called the **learning rate**(s).

### D. Results

The code was run for 100 iterations and the best performing netlists were handpicked. By best performing netlists, here I mean that the netlists whose loss function was less than a specified threshold. This threshold was determined by running the code several times. This is a **tunable hyperparameter** of the model used. Out of all the best performing netlists, the netlist that had the least sensitivity was handpicked and its properties have been listed in Table V.

| Requirement | Specification | Starting Point | Final Result |
|---|---|---|---|
| Gain | 10 dB | 9.11 dB | 12.114 dB |
| Noise Figure | 4 dB | 4.80 dB | 3.8133 dB |
| $S_{11}$ | -10 dB | -12.94 dB | -18.8709 dB |
| $IIP_3$ | -15 dBm | -10.35 dBm | -13.43 dBm |
| Power | Minimize | 1.288 mW | 0.95 mW |
| Loss | | 1.6857 | 0.0010 |

TABLE V
RESISTIVE FEEDBACK LNA - RESULTS

The temperature analysis for the best netlist is given below in Table VI.

| Temperature | Loss |
|---|---|
| -40°C | 0.198546 |
| 0°C | 0.0008786 |
| 27°C | 0.0009422 |
| 100°C | 0.366607 |

TABLE VI
TEMPERATURE VARIATIONS

The process analysis for the best netlist is given below in Table VII.

| Process | Loss |
|---|---|
| TOP TT | 0.0009422 |
| TOP FF | 0.0013185 |
| TOP FS | 0.00086715 |
| TOP SF | 0.00103126 |
| TOP SS | 0.43451564 |

TABLE VII
PROCESS VARIATIONS

The supply voltage analysis for the best netlist is given below in Table VIII

| Supply Voltage | Loss |
|---|---|
| 0.81 V | 0.042939156 |
| 0.90 V | 0.0009421553 |
| 0.99 V | 0.001157762 |

TABLE VIII
SUPPLY VOLTAGE VARIATIONS

The sensitivity was evaluated to be 1.62985x10$^{-5}$ while the temperature sensitivity evaluated to be 5.328x10$^{-7}$. The time taken to run this code was about 30 minutes.
The final parameters of the best netlist came out to be $m_1$=26, $m_2$=86, $R_f$=302 Ω, $I_b$=3.99uA, $L_s$ = 1pH and $L_g$ = 100nH. The number of netlists that passed the Gradient Descent Loss Threshold were 12, the number of netlists that passed the PVT check were 3 and the number of netlists that passed the sensitivity check were 3 over 100 iterations. The lowest loss obtained was 0.0010 and sensitivity of the same was 1.6298x10$^{-5}$.

### *E. Loss Function*

As anticipated the loss function kept on decreasing as the number of iterations increased. After sometime, it increased and then decreased again as seen in Figure 6. This indicates that the algorithm used is converging to the local minimum solution as required. There are two local minima as apparent and either one is acceptable. The loss function used in this case was the simple Rectified Linear Unit (ReLU) loss function defined in equation (1) The total loss was a sum total of the losses corresponding to the Gain, $S_{11}$, $IIP_3$, $I_{dd}$ and Noise Figure (NF) as described in Section II.

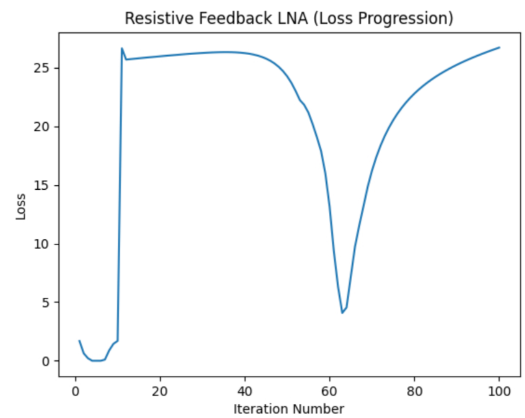


Fig. 6. Loss progression in Resistive Feedback LNA

## V. CONCLUSION AND DISCUSSION

The role of a designer is crucial and cannot be replaced. This automation makes the design process only faster and easier. The local minima highly depends on the choice of initial netlist. The "better" it is in relation to the optimum, the faster it converges. The key drawback in this type of automation is that, the code cannot generate the netlists, it just makes the existing ones better.

## REFERENCES


[1] F. Bruccoleri, E. Klumperink, and B. Nauta. Noise cancelling in wideband cmos lnas. In *2002 IEEE International Solid-State Circuits Conference. Digest of Technical Papers (Cat. No.02CH37315)*, volume 1, pages 406–407 vol.1, 2002. doi: 10.1109/ISSCC.2002.993104.

[2] S. Devi, G. Tilwankar, and R. Zele. Automated design of analog circuits using machine learning techniques. In *2021 25th International Symposium on VLSI Design and Test (VDAT)*, pages 1–6, 2021. doi: 10.1109/VDAT53777.2021.9601131.

[3] S. Joo, T.-Y. Choi, and B. Jung. A 2.4-ghz resistive feedback lna in 0.13-$\mu$m cmos. *IEEE Journal of Solid-State Circuits*, 44(11):3019–3029, 2009. doi: 10.1109/JSSC.2009.2031912.

[4] K. Settaluri, Z. Liu, R. Khurana, A. Mirhaj, R. Jain, and B. Nikolic. Automated design of analog circuits using reinforcement learning. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 41(9): 2794–2807, 2022. doi: 10.1109/TCAD.2021.3120547.

[5] N. Yadav, A. Pandey, and V. Nath. Design of cmos low noise amplifier for 1.57ghz. In *2016 International Conference on Microelectronics, Computing and Communications (MicroCom)*, pages 1–5, 2016. doi: 10.1109/MicroCom.2016.7522438.